

DETAILED ACTION

1. A request for continued examination under 37 CFR 1.114, including the fee set forth in 37 CFR 1.17(e), was filed in this application after final rejection. Since this application is eligible for continued examination under 37 CFR 1.114, and the fee set forth in 37 CFR 1.17(e) has been timely paid, the finality of the previous Office action has been withdrawn pursuant to 37 CFR 1.114.

Applicant's submission filed on November 12, 2009 has been entered.

2. Claims 1-8, 10-28, and 30-42 have been examined.

Response to Amendments

3. In the instant amendment, claims 1, 21, 41, and 42 have been amended; and claims 9 and 29 have been canceled.

4. The objection to claims 1, 21, 24, 27, and 41 is withdrawn in view of Applicant's amendments.

Response to Arguments

5. Applicants' arguments have been fully considered. After further consideration, examiner notes that the combination of Jensen, Kjellberg, and Krantz also teaches the newly added limitations as set forth in details below.

Furthermore, although Jensen has been still relied upon as the primary reference, a new ground of rejection has been applied to reject particular newly added claimed limitations

accordingly by the following method: a set of provisioning instructions is stored in the content type, the provisioning instructions being customized by distributed provisioning control through the provisioning instructions for different versions of the application and being coupled to the application for enabling a provisioning

Accordingly, Applicants' arguments are moot.

Claim Rejections – 35 USC §103

6. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

7. Claims 1-8, 10-28, and 30-42 are rejected under 35 U.S.C. 103(a) as being unpatentable over Jensen (art of record, US Patent No. 2004/0261086 A1 in view of Kjellberg (art of record, US Patent No. 2003/0084165 A1) and Krantz (art of record, US Patent No. 2005/0091357 A1).

Claim 1:

Jensen discloses *a method for providing customized provisioning of an application on a runtime environment of a terminal* (e.g., FIG. 2, provisioning digital services/applications and deploying/installing the provisioned digital services/applications to target devices 202a-c, [0014]-[0015], [0024]-[0030]),

the application including content (e.g., Provisioning Application 208, Database 220, and digital services/applications include a plurality of contents, [0025], [0033]-[0035]),

having at least one specified content type (e.g., specified content type such as target devices/user profiles, price/service plans, [0027]-[0033], [0037]-[0042]), *the method comprising the steps of:*

for each content type, obtaining the content by the runtime environment (e.g., FIG. 2, retrieving/obtaining target devices/user profiles and price/service plans by runtime environment of Provisioning Server 204, [0024]-[0027], [0031]-[0033]);

obtaining by the runtime environment a set of provisioning instructions related to the content type (e.g., [0027][0033], for each device/user profile, price/service

plan → executing/obtaining a set or provisioning instructions associated with said device/user profile, price/service plan, [0037]-[0042]),

the provisioning instructions being customized (e.g., [0032] and [0035], for particular device/user profile and identified price/service plan → specific provisioning instructions are executed, i.e., customized)

by distributed provisioning control through the provisioning instructions (e.g., FIG. 2, execution/control of provisioning instructions and/or provisioning APIs have been distributed over Provisioning Application 208, Provisioning API 222, Adapter 206, but not been hardcoded in target devices 202),

the provisioning instructions coupled to the application for specifying a provisioning Application Program Interface (API) set for provisioning the content on the terminal (e.g., FIG. 3, Provisioning Server 204 specifying either Discovery, Subscription, or Delivery Provisioning API set, [0028]-[0031], [0036]-[0039], [0041]-[0043]); *and*

executing by the runtime environment the provisioning instructions for employing the API set to provision the application according to the specified content type (e.g., FIG. 2, provisioning digital services/applications according to target device/user profiles and/or price/service plans, [0014]-[0015], [0043]-[0047], [0050]-[0053]).

Jensen does not explicitly disclose *[the provisioning instructions being customized]* for different subsets of versions of the application.

However, in an analogous art, Kjellbert further discloses *[the provisioning instructions being customized]* for different subsets of versions of the application (e.g., [0024]-[0026]).

It would have been obvious to a person having ordinary skill in the art at the time the invention was made to combine Kjellberg's teaching into Jensen's teaching. One would have been motivated to do so to provision a suitable/new version to client devices in real-time as suggested by Kjellberg (e.g., [0025]-[0026]).

Neither Jensen nor Kjellberg explicitly discloses *[executing by the runtime environment the provisioning instructions for employing the API set,] by a script interpreter.*

However, in an analogous art, Krantz further discloses *[executing by the runtime environment the provisioning instructions for employing the API set,] by a script interpreter (e.g., [0089], XML parser/interpreter as a script interpreter).*

It would have been obvious to a person having ordinary skill in the art at the time the invention was made to combine Krantz's teaching into Jensen and Kjellberg's teaching. One would have been motivated to do so to provide network provisioning services by using XML rules files, configuration files, and provisioning files as suggested by Krantz (e.g., [0047], [0063], and [0089]).

Claim 2:

Jensen discloses *the method according to claim 1, wherein provisioning control of the content is shared between the runtime environment and the application through the coupled provisioning instructions (e.g., [0025], [0033]-[0035]).*

Claim 3:

Jensen discloses *the method according to claim 2 further comprising the step of employing a provisioning service to direct the provisioning API, the service configured for recognizing the provisioning instructions (e.g., [0014]-[0015], [0024]-[0030]).*

Claim 4:

Jensen discloses *the method according to claim 3 further comprising the step of the service customizing the provisioning process and the associated provisioning API set according to the provisioning instructions (e.g., [0024]-[0027], [0031]-[0033]).*

Claim 5:

Jensen discloses *the method according to claim 4, wherein the service is shared by a plurality of the applications (e.g., [0028]-[0031], [0036]-[0039], [0041]-[0043]).*

Claim 6:

Jensen discloses *the method according to claim 3 further comprising the step of employing a standard one of the provisioning API set by the service (e.g., [0026]-[0030], [0040]-[0042]).*

Claim 7:

Jensen discloses *the method according to claim 6 further comprising the step of obtaining remotely a custom API via a network coupled to the terminal (e.g., [0031]-[0034], [0043]-[0046]).*

Claim 8:

Jensen discloses *the method according to claim 2, wherein the provisioning instructions are selected from the group comprising code, script, and configuration data (e.g., [0027]-[0033], [0037]-[0042]).*

Claim 10:

Jensen discloses *the method according to claim 8, wherein the provisioning instructions are separate from the content (e.g., [0014]-[0015], [0024]-[0030]).*

Claim 11:

Jensen discloses *the method according to claim 10 further comprising the step of accessing the provisioning instructions remotely from the terminal (e.g., [0014]-[0015], [0043]-[0047], [0050]-[0053]).*

Claim 12:

Jensen discloses *the method according to claim 11, wherein the remote access of the provisioning instructions is in conjunction with a networked repository (e.g., [0025], [0033]-[0035]).*

Claim 13:

Jensen discloses *the method according to claim 12, wherein the terminal is selected from the group comprising wired devices and wireless devices (e.g., [0027]-[0033], [0037]-[0042]).*

Claim 14:

Jensen discloses *the method according to claim 5, wherein a generic API is included in the provisioning API set, the generic API configured for addressing by at least two dissimilar ones of the specified content type (e.g., [0026]-[0030], [0040]-[0042]).*

Claim 15:

Jensen discloses *the method according to claim 14 further comprising the step of employing a series of enablers for providing access to corresponding selected ones of the generic API, each of the enablers associated with a predefined content type (e.g., [0024]-[0027], [0040]-[0042], [0047]-[0050]).*

Claim 16:

Jensen discloses *the method according to claim 2, wherein a generic API is included in the provisioning API set, the generic API configured for addressing by at least two dissimilar ones of the specified content type (e.g., [0031]-[0034], [0043]-[0046]).*

Claim 17:

Jensen discloses *the method according to claim 16 further comprising the step of employing a series of enablers for providing access to corresponding selected ones of the generic API, each of the enablers associated with a predefined content type (e.g., [0027]-[0033], [0041]-[0043]).*

Claim 18:

Jensen discloses *the method according to claim 17, wherein the enabler is an executable unit that executes provisioning instruction requests for the predefined content type* (e.g., [0028]-[0031], [0036]-[0039], [0041]-[0043]).

Claim 19:

Jensen discloses *the method according to claim 18 further comprising the step of obtaining the enabler selected from the group comprising: locally on the terminal by a provisioning service* (e.g., [0025], [0033]-[0035]);

bundled with a content descriptor of the content; and remotely from the terminal by the provisioning service (e.g., [0024]-[0027], [0031]-[0033]).

Claim 20:

Jensen discloses *the method according to claim 5, wherein the provisioning instructions were amended prior to the step of obtaining the provisioning instructions by the runtime environment* (e.g., [0014]-[0015], [0024]-[0030]).

Claim 21:

Jensen discloses *a terminal, including a computer processor and a computer readable storage medium for providing customized provisioning of an application on a runtime environment* (e.g., FIG. 2, provisioning applications/services from Provisioning Application 208 and Database 220, and deploying/installing provisioned applications/services on target devices 202a-c, [0014]-[0015], [0024]-[0030]),

the application including content (e.g., Provisioning Application 208 and Database 220 includes a plurality of contents, [0025], [0033]-[0035])

having at least one specified content type (e.g., target devices/user profiles, price/service plans, [0027]-[0033], [0037]-[0042]), *the terminal comprising:*

a processing framework for obtaining the content (e.g., FIG. 2, runtime environment of Provisioning Server 204, [0024]-[0027], [0031]-[0033]);

obtaining by the runtime environment a set of provisioning instructions related to the content type (e.g., [0027][0033], for each profile, price/service plan →

executing/obtaining a set or provisioning instructions associated with said profile, price/service plan, [0037]-[0042]),

the provisioning instructions being customized (e.g., [0032] and [0035], for particular client device data and identified price/service plan → specific provisioning instructions are executed, i.e., customized)

by distributed provisioning control through the provisioning instructions (e.g., FIG. 2, execution/control of provisioning instructions/APIs have been distributed over Provisioning Application 208, Provisioning API 222, Adapter 206, but not been hardcoded in target devices 202),

a provisioning API set included in the processing framework for provisioning the content (e.g., FIG. 3, Provisioning Server 204 specifying either Discovery, Subscription, or Delivery Provisioning API set, [0028]-[0031], [0036]-[0039], [0041]-[0043]); and

a set of provisioning instructions related to the content, the provisioning instructions coupled to the application for specifying selected ones of the provisioning API set (e.g., FIG. 2, provisioning applications/services according to target devices/user profiles/price plans, [0014]-[0015], [0043]-[0047], [0050]-[0053]).

Jensen does not explicitly disclose *the provisioning instructions being customized for different subsets of versions of the application*.

However, in an analogous art, Kjellbert further discloses *the provisioning instructions being customized for different subsets of versions of the application* (e.g., [0024]-[0026]).

It would have been obvious to a person having ordinary skill in the art at the time the invention was made to combine Kjellberg's teaching into Jensen's teaching. One would have been motivated to do so to provision a suitable/new version to client devices in real-time as suggested by Kjellberg (e.g., [0025]-[0026]).

Neither Jensen nor Kjellberg explicitly discloses *executing by the runtime environment the provisioning instructions for employing the API set, by a script interpreter*.

However, in an analogous art, Krantz further discloses *executing by the runtime environment the provisioning instructions for employing the API set, by a script interpreter* (e.g., [0089], XML parser/interpreter as a script interpreter).

It would have been obvious to a person having ordinary skill in the art at the time the invention was made to combine Krantz's teaching into Jensen and Kjellberg's teaching. One would have been motivated to do so to provide network provisioning services by using XML rules files, configuration files, and provisioning files as suggested by Krantz (e.g., [0047], [0063], and [0089]).

Claim 22:

Jensen discloses *the terminal according to claim 21, wherein provisioning control of the content is shared between the framework and the application through the coupled provisioning instructions* (e.g., [0027]-[0033], [0037]-[0042]).

Claim 23:

Jensen discloses *the terminal according to claim 22 further comprising a provisioning service to direct the provisioning API, the service configured for recognizing the provisioning instructions* (e.g., [0028]-[0031], [0036]-[0039], [0041]-[0043]).

Claim 24:

Jensen discloses *the terminal according to claim 23 wherein the service is configured for customizing the provisioning process and the associated provisioning API set according to the provisioning instructions* (e.g., [0014]-[0015], [0024]-[0030]).

Claim 25:

Jensen discloses *the terminal according to claim 24, wherein the service is shared by a plurality of the applications* (e.g., [0014]-[0015], [0043]-[0047], [0050]-[0053]).

Claim 26:

Jensen discloses *the terminal according to claim 23, wherein the service employs a standard one of the provisioning API set* (e.g., [0025], [0033]-[0035]);

Claim 27:

Jensen discloses *the terminal according to claim 26, a custom API is obtained remotely by the service via a network coupled to the terminal* (e.g., [0024]-[0027], [0031]-[0033]).

Claim 28:

Jensen discloses *the terminal according to claim 22, wherein the provisioning instructions are selected from the group comprising code, script, and configuration data* (e.g., [0026]-[0030], [0040]-[0042]).

Claim 30:

Jensen discloses *the terminal according to claim 28, wherein the provisioning instructions are separate from the content* (e.g., [0024]-[0027], [0040]-[0042], [0047]-[0050]).

Claim 31:

Jensen discloses *the terminal according to claim 30, wherein the provisioning instructions are configured for obtaining the remotely from the terminal* (e.g., [0031]-[0034], [0043]-[0046]).

Claim 32:

Jensen discloses *the terminal according to claim 31, wherein the remote access of the provisioning instructions is in conjunction with a networked repository (e.g., [0014]-[0015], [0033]-[0035]).*

Claim 33:

Jensen discloses *the terminal according to claim 32, wherein the terminal is selected from the group comprising wired devices and wireless devices (e.g., [0027]-[0033], [0041]-[0043]).*

Claim 34:

Jensen discloses *the terminal according to claim 25, wherein a generic API is included in the provisioning API set, the generic API configured for addressing by at least two dissimilar ones of the specified content type (e.g., [0026]-[0030], [0040]-[0043]).*

Claim 35:

Jensen discloses *the terminal according to claim 34 further comprising a series of enablers for providing access to corresponding selected ones of the generic API, each of the enablers associated with a predefined content type (e.g., [0014]-[0015], [0043]-[0047], [0050]-[0053]).*

Claim 36:

Jensen discloses *the terminal according to claim 22, wherein a generic API is included in the provisioning API set, the generic API configured for addressing by at least two dissimilar ones of the specified content type (e.g., [0014]-[0015], [0024]-[0030]).*

Claim 37:

Jensen discloses *the terminal according to claim 36 further comprising a series of enablers for providing access to corresponding selected ones of the generic API,*

each of the enablers associated with a predefined content type (e.g., [0027]-[0033], [0037]-[0042]).

Claim 38:

Jensen discloses the terminal according to claim 37, wherein the enabler is an executable unit that executes provisioning instruction requests for the predefined content type (e.g., [0024]-[0027], [0031]-[0033]).

Claim 39:

Jensen discloses the terminal according to claim 38, wherein the enabler location is selected from the group comprising: locally on the terminal by a provisioning service (e.g., [0028]-[0031], [0036]-[0039], [0041]-[0043]);

bundled with a content descriptor of the content; and remotely from the terminal by the provisioning service (e.g., [0025], [0033]-[0035]).

Claim 40:

Jensen discloses the terminal according to claim 25, wherein the provisioning instructions were amended prior to the step of obtaining the provisioning instructions by the runtime environment (e.g., [0026]-[0030], [0040]-[0042]).

Claim 41:

Jensen discloses a method for providing customized provisioning of an application on a runtime environment of a terminal (e.g., FIG. 2, provisioning applications/services from Provisioning Application 208 and Database 220, and deploying/installing provisioned applications/services on target devices 202a-c, [0014]-[0015], [0024]-[0030]),

the application including content (e.g., Provisioning Application 208 and Database 220 includes a plurality of contents, [0025], [0033]-[0035])

having at least one specified content type (e.g., target devices/user profiles, price/service plans, [0027]-[0033], [0037]-[0042]), the method comprising the steps of:

sending the content for receipt by the runtime environment (e.g., FIG. 2, runtime environment of Provisioning Server 204, [0024]-[0027], [0031]-[0033]);

obtaining by the runtime environment a set of provisioning instructions related to the content type (e.g., [0027][0033], for each profile, price/service plan → executing/obtaining a set or provisioning instructions associated with said profile, price/service plan, [0037]-[0042]),

the provisioning instructions being customized (e.g., [0032] and [0035], for particular client device data and identified price/service plan → specific provisioning instructions are executed, i.e., customized)

by distributed provisioning control through the provisioning instructions (e.g., FIG. 2, execution/control of provisioning instructions/APIs have been distributed over Provisioning Application 208, Provisioning API 222, Adapter 206, but not been hardcoded in target devices 202),

sending a set of provisioning instructions related to the content for receipt by the runtime environment, the provisioning instructions coupled to the application for specifying a provisioning API set for provisioning the content (e.g., FIG. 3, Provisioning Server 204 specifying either Discovery, Subscription, or Delivery Provisioning API set, [0028]-[0031], [0036]-[0039], [0041]-[0043]); and

making available selected ones of the API provisioning set for use by the provisioning instructions; wherein execution of the provisioning instructions employs the API provisioning set to provision the application according to the specified content type (e.g., FIG. 2, provisioning applications/services according to target devices/user profiles/price plans, [0014]-[0015], [0043]-[0047], [0050]-[0053]).

Jensen does not explicitly disclose *the provisioning instructions being customized for different subsets of versions of the application.*

However, in an analogous art, Kjellbert further discloses *the provisioning instructions being customized for different subsets of versions of the application* (e.g., [0024]-[0026]).

It would have been obvious to a person having ordinary skill in the art at the time the invention was made to combine Kjellberg's teaching into Jensen's teaching. One would have been motivated to do so to provision a suitable/new version to client devices in real-time as suggested by Kjellberg (e.g., [0025]-[0026]).

Neither Jensen nor Kjellberg explicitly discloses *executing by the runtime environment the provisioning instructions for employing the API set, by a script interpreter*.

However, in an analogous art, Krantz further discloses *executing by the runtime environment the provisioning instructions for employing the API set, by a script interpreter* (e.g., [0089], XML parser/interpreter as a script interpreter).

It would have been obvious to a person having ordinary skill in the art at the time the invention was made to combine Krantz's teaching into Jensen and Kjellberg's teaching. One would have been motivated to do so to provide network provisioning services by using XML rules files, configuration files, and provisioning files as suggested by Krantz (e.g., [0047], [0063], and [0089]).

Claim 42:

Jensen discloses *a computer program product for providing customized provisioning of an application on a runtime environment of a terminal* (e.g., FIG. 2, provisioning applications/services from Provisioning Application 208 and Database 220, and deploying/installing provisioned applications/services on target devices 202a-c, [0014]-[0015], [0024]-[0030]),

the application including content (e.g., Provisioning Application 208 and Database 220 includes a plurality of contents, [0025], [0033]-[0035])

having at least one specified content type (e.g., target devices/user profiles, price/service plans, [0027]-[0033], [0037]-[0042]), the computer program product comprising:

a computer readable medium; a processing framework module stored on the computer readable medium for obtaining the content (e.g., FIG. 2, runtime environment of Provisioning Server 204, [0024]-[0027], [0031]-[0033]);

a provisioning service module coupled to the framework module, the provisioning service module configured for utilizing a provisioning API set for provisioning the content (e.g., FIG. 3, Provisioning Server 204 specifying either Discovery, Subscription, or Delivery Provisioning API set, [0028]-[0031], [0036]-[0039], [0041]-[0043]);

obtaining by the runtime environment a set of provisioning instructions related to the content type (e.g., [0027][0033], for each profile, price/service plan → executing/obtaining a set or provisioning instructions associated with said profile, price/service plan, [0037]-[0042]),

the provisioning instructions being customized (e.g., [0032] and [0035], for particular client device data and identified price/service plan → specific provisioning instructions are executed, i.e., customized)

by distributed provisioning control through the provisioning instructions (e.g., FIG. 2, execution/control of provisioning instructions/APIs have been distributed over Provisioning Application 208, Provisioning API 222, Adapter 206, but not been hardcoded in target devices 202), and

an interpreter module coupled to the framework module, the interpreter module configured for executing a set of provisioning instructions related to the content (e.g., FIG. 2, provisioning applications/services according to target devices/user profiles/price plans, [0014]-[0015], [0043]-[0047], [0050]-[0053]),

the provisioning instructions associated with the application for specifying selected ones of the provisioning API set (e.g., FIG. 3, Provisioning Server 204 specifying either Discovery, Subscription, or Delivery Provisioning API set, [0028]-[0031], [0036]-[0039], [0041]-[0043]).

Jensen does not explicitly disclose *the provisioning instructions being customized for different subsets of versions of the application*.

However, in an analogous art, Kjellbert further discloses *the provisioning instructions being customized for different subsets of versions of the application* (e.g., [0024]-[0026]).

It would have been obvious to a person having ordinary skill in the art at the time the invention was made to combine Kjellberg's teaching into Jensen's teaching. One would have been motivated to do so to provision a suitable/new version to client devices in real-time as suggested by Kjellberg (e.g., [0025]-[0026]).

Neither Jensen nor Kjellberg explicitly discloses *executing by the runtime environment the provisioning instructions for employing the API set, by a script interpreter*.

However, in an analogous art, Krantz further discloses *executing by the runtime environment the provisioning instructions for employing the API set, by a script interpreter* (e.g., [0089], XML parser/interpreter as a script interpreter).

It would have been obvious to a person having ordinary skill in the art at the time the invention was made to combine Krantz's teaching into Jensen and Kjellberg's teaching. One would have been motivated to do so to provide network provisioning services by using XML rules files, configuration files, and provisioning files as suggested by Krantz (e.g., [0047], [0063], and [0089]).

Conclusion

8. Any inquiry concerning this communication should be directed to examiner Thuy (Twee) Dao, whose telephone/fax numbers are (571) 272 8570 and (571) 273 8570, respectively. The examiner can normally be reached on every Tuesday, Thursday, and Friday from 6:00AM to 6:00PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Tuan Q. Dam, can be reached at (571) 272 3695.

Any inquiry of a general nature of relating to the status of this application or proceeding should be directed to the TC 2100 Group receptionist whose telephone number is (571) 272 2100.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

/Twee Dao/
Examiner, Art Unit 2192

/Tuan Q. Dam/
Supervisory Patent Examiner, Art Unit 2192